Research Article

Clustered Exact Daum-Huang Particle Flow Filter

Halit Örenbaş ⬡ and Muharrem Mercimek

Control and Automation Engineering, Yildiz Technical University, Istanbul 34220, Turkey

Correspondence should be addressed to Halit Örenbaş; orenbas@yildiz.edu.tr

Unlike the conventional particle filters, particle flow filters do not rely on proposal density and importance sampling; they employ flow of the particles through a methodology derived from the log-homotopy scheme and ensure successful migration of the particles. Amongst the efficient implementations of particle filters, Exact Daum-Huang (EDH) filter pursues the calculation of migration parameters all together. An improved version of it, Localized Exact Daum-Huang (LEDH) filter, calculates the migration parameters separately. In this study, the main objective is to reduce the cost of calculation in LEDH filters which is due to exhaustive calculation of each migration parameter. We proposed the Clustered Exact Daum-Huang (CEDH) filter. The main impact of CEDH is the clustering of the particles considering the ones producing similar errors and then calculating the same migration parameters for the particles within each cluster. Through clustering and handling the particles with high errors, their engagement and influence can be balanced, and the system can greatly reduce the negative effects of such particles on the overall system. We implement the filter successfully for the scenario of high dimensional target tracking. The results are compared to those obtained with EDH and LEDH filters to validate its efficiency.

1. Introduction

For the analysis, inference, and comprehensive understanding of a dynamic system, two models are essentially required. The first model is the system model, which describes the change of its states with respect to time, and embodies the information to characterize the entire system. The second model is the measurement model which represents the noise measurements associated with the state vectors.

The probabilistic state-space representations of these models and the need for the information update upon having new measurements are preferably consistent with the practical Bayesian approach [1]. State-space approach provides great advantages in handling the multivariable nonlinear processes over conventional time series techniques [2].

Bayesian approach constructs a posterior probability density function of a state based on available information for the purpose of dynamic state estimation. The use of a recursive filter can be a good solution; in that, an estimation takes place when a new measurement is obtained. The received data is preferably processed sequentially rather than being processed as a whole in the recursive filter approach. Therefore, there is

no need to store the complete data and use it when estimation is required upon receiving an instant measurement. The prediction and update stages must be employed for state estimation. At the prediction stage, the system model is used to construct the pdf of the state for the next measurement time. The state is deteriorated due to unknown disturbances; hence, during the prediction stage, the pdf of the state is mostly deformed or distorted. The update stage employs the Bayes theorem to modify the prediction pdf. Thus, the latest measurements establish the comprehension of the target state [1].

Particle filters are the sequential Monte Carlo algorithms which can be applied to any state-space model [3]. They are the generalized form of the Kalman filters. Particle representations of the probability densities are established with particle filters. The Bootstrap particle filter (BPF) updates the weight of each particle after drawing the particles from the prior distribution using the recent measurement likelihoods [4]. If salient measurements are employed or the dimension of the state is high, then many particles from the prior distribution will be in regions with low likelihood. This surely

results in assigning low weights to many particles in BPF [5]. Hence Monte Carlo approximation for the posterior distribution is established depending on too few particles; the degeneracy due to weights leads to faulty posterior distribution depictions [2, 3].

The variance of the importance weights can be minimized using the optimal proposal distribution approximation [6]. Cutting-edge particle filters mimic the well-known strategy of the optimal proposal distribution approximation through establishing proposal distributions in efficiency [7]. Auxiliary particle filters (APF) were put forward to sample the particles more appropriately considering the new measurements in hand [8]. The variances of Monte Carlo estimates are reduced with Rao-Blackwellised particle filters through isolating some of the states wisely [9]. The unscented transformation aiming to employ an error-bounded deterministic sampling method is adopted in the unscented particle filter [10]. The approaches mentioned so far can be conveniently used under certain circumstances within a certain range of problem setting [11]. Particle filter use for the states in high dimensional space involved with the scenario of multitarget tracking still is a challenging task [12]. Conventional particle filter methods can be so naive to handle high dimensionality problem [13]. There are promising methods providing good performance, such as a well-known method, the equivalent weights particle filter [14]. It fights against the weight degeneracy but lacks the statistical consistency of the conventional particle filters. The separation of the state space through partitioning or segmenting has been adopted in some studies [15, 16]. The technique relies essentially on the factorization of the conditional posterior; hence it has a narrow scope of application. An all-intentionally approach is the use of Markov Chain Monte Carlo (MCMC) method to aid the particle filters [17]. The number of particles and consequently the cost of calculation grow exponentially when higher state dimensions are dealt with [18].

An unconventional strategy was proposed by Daum and Huang as a nonlinear filtering approach [19–21]. A homotopy between the logarithms of unnormalized prior and posterior distributions is presented at each step of Daum-Huang (DH) filters. The homotopy depicts the particle flow which can be regarded as the solution to a partial differential equation. The particle flow can be stated to be guaranteeing the gradual migration of the particles to the regions with higher posterior [18].

Extraordinary performance is reported to be obtained when DH filters are utilized in a number of nonlinear filtering systems. On the other hand, the researchers also report that the unqualified dependence of EDH filter on the extended or unscented Kalman filters (EKF/UKF), which are executed in parallel, yields performance drawbacks [22]. A solution to this is proposed in [18] through focusing on the cases that the system and the observation noise are both in the Gaussian form, the system map is linear, but the observations are in a highly nonlinear manner. The modified version is accordingly proposed and introduced as LEDH. The modification actually gives support in the linearization of the system and the calculation of each migration parameter of each particle individually rather than dealing with a single calculation associating all particles. It is not difficult to state that the cost of calculation is the weakness of this eminent filter despite the fact that it has a remarkably good performance.

Standard particle filters utilizing importance weights have been reported to be struggling with exponentially increasing sample size which is essentially due to high dimension of the state space. Therefore, particle filtering inherits weight degeneracy in case of having high dimensional filtering scenarios. The problems arising from high dimensionality have been discussed in the literature regarding different aspects.

Recent studies on particle flow filters provide solutions to the problems arising from weight degeneracy. Li et al. present new filters which integrate deterministic particle flows into a particle filter framework [23]. The proposed theoretical scheme can provide the adequacy of the particle filter. Also it can sustain the efficiency of particle flow methods. Li and Coates strive the computational burden of the particle flow particle filter by incorporating clustering of the particles [24]. In this study, we deal solely with improving the efficiency of particle flow filter which is inherently easier to implement and thus more suitable in practical scenarios. The main impact of CEDH is the clustering of the particles considering the ones producing similar errors and then calculating the same migration parameters for the particles within each cluster. Through clustering and handling the particles with high errors, their engagement and influence can be balanced, and the system can greatly reduce the negative effects of such particles on the overall system.

Surace et al. [25] focus on the varying aspects of the curse of dimensionality in continuous time filtering. They investigate the use of optimal feedback control scheme that deals with importance weights. Daum et al. [26] extended their studies through derivation of a new exact stochastic particle flow filter using a theorem established by Gromov. They conducted numerical experiments for a number of different high dimensional problems. In [27], the researchers combined the strengths of invertible particle flow and sequential Markov chain Monte Carlo (SMCMC) through constructing a composite Metropolis-Hastings (MH) kernel. They also proposed a Gaussian mixture model- (GMM-) based particle flow algorithm to construct effective MH kernels.

Our main objective is to reduce and balance the influence of particles which produce high errors through clustering in Particle Flow Filter. This will provide reduced cost of calculation while maintaining the performance of LEDH. The error value for each particle is taken into account and k-means++ algorithm [28] is employed for clustering. Such a clustering provides a significant benefit in the calculation of migration parameters of the particles which are designated to be in the same cluster.

Unlike the EDH filter which calculates of migration parameters all together and LEDH filter which calculates the migration parameters separately, the same migration parameters are pursued for the particles in the same cluster. The number of the migration parameters accompanying each cluster can be reduced and this yields an ease on the calculations of the LEDH.

In this study for each target velocity, position, derivative of the velocity, and derivative of the position are dealt with; thus we are involved with a tracking problem in the four-dimensional state space. Four targets are tracked utilizing our CEDH filter as well as EKF, BPF, EDH, and LEDH filters and performances are compared.

The structure of this paper is as follows. In Section 2 theoretical base in terms of established techniques for particle flow filters is represented. The implementation details of k-means++ algorithm are given and its use in CEDH filter is described in Section 3. The application of CEDH filter on a descriptive scenario is detailed in Section 4. Performance comparisons of the filters employed for multitarget tracking problem in the high dimensional state space are given in this section as well. Section 5 puts concluding comments on the proposed filter, discusses problems that arose in the study and possible future work, and lists practical pieces of advice resulting from the scenario.

2. Particle Flow Filter

Particle flow filter is the modified version of the conventional particle filters. It essentially fights against the particle degeneracy and ensures the fast convergence to the particles with the highest posterior distribution in the next step through employing a logarithmic homotopy function. The homotopy function depicts the transition between prior and posterior distributions for the flow of particles. The filter uses the Ito stochastic Partial Differential Equation (PDE) which is used to differentiate between the step parameter of the homotopy function and the state.

We consider a discrete-time nonlinear filtering task with the following models:

$$x_k = f_k(x_{k-1}) + v_k \tag{1}$$

$$z = \gamma(x_{k-1}) + n_k \tag{2}$$

where $x_k \in \mathbb{R}^D$ is a target state vector, $z_k \in \mathbb{R}^S$ is a measurement vector, $v_k \in \mathbb{R}^D$ is the process noise, and $n_k \in \mathbb{R}^S$ is the measurement noise. $f_k : \mathbb{R}^D \longrightarrow \mathbb{R}^D$ is a nonlinear map and $\gamma_k : \mathbb{R}^D \longrightarrow \mathbb{R}^S$ is a nonlinear measurement map.

Bayesian rule to define the unnormalized marginal posterior distribution is as follows:

$$\widetilde{p}(x_k \mid z_{1:k}) = p(z_k \mid x_k) p(x_k \mid z_{1:k-1}). \tag{3}$$

Daum and Huang expressed the homotopy function in this form:

$$\phi(x_k, \lambda) = \log g(x_k) + \lambda \log h(x_k) \tag{4}$$

where $h(x_k) = p(z_k \mid x_k)$, $g(x_k) = p(x_k \mid z_{1:k-1})$, and λ is the real valued step parameter in the range of $[0, 1]$ and represents the intensity or amount of particle flow. The homotopy function provides a continuous deformation from $\log g(x_k)$ (when $\lambda = 0$) to the logarithm of the unnormalized posterior distribution, $\log \widetilde{p}(x_k \mid z_{1:k})$ (when $\lambda = 1$).

In the original particle flow filter, the flow is improved so that the homotopy function remains constant as λ evolves.

Partial differential equation use is required for this purpose, and the following equation can be referred to solve and calculate the flow of the particles:

$$\frac{\partial \varphi}{\partial x} \frac{dx}{d\lambda} + \frac{\partial \varphi}{\partial \lambda} = 0. \tag{5}$$

Daum and Huang derived the following expression utilizing the Fokker-Planck equation and developed and generalized the filter through proposing EDH filter [21]:

$$\frac{\partial \varphi}{\partial x} \psi(x, \lambda) + \log(h) = -Tr\left(\frac{\partial \psi}{\partial x}\right). \tag{6}$$

The solution of this equation leads to the exact flow of the probability density. The flow quantities of the particles can be calculated as given in [21]:

$$\frac{dx}{d\lambda} = A(\lambda) x + b(\lambda) \tag{7}$$

where

$$A = -\frac{1}{2} P H^T \left(\lambda H P H^T + R\right)^{-1} H \tag{8}$$

and

$$b = (I - 2\lambda A)\left[(I + \lambda A) P H^T R^{-1} z + A\overline{x}\right]. \tag{9}$$

P and R depict the covariance of the estimation error and measurement noise, respectively. H is the measurement matrix, x denotes the state variable that immigrates in each cycle, and $dx/d\lambda$ represents the change of x with respect to λ.

2.1. Implementation of Particle Flow Filter. Algorithm 1 includes the pseudocodes of the EDH filter implementation regarding the previously stated theoretical base. Particle migration task is realized following the steps on lines 7 to 16.

N is the number of particles, and T is the number of time intervals. The UKF/EKF state and covariance matrix estimation are represented with m and P, respectively.

The steps of UKF/EKF estimation and UKF/EKF update are expressed on lines 6 and 17, respectively. The measurement matrix $H_{\overline{x}}$ is calculated through linearizing the current estimate $\overline{x}_k$ and depicted on line 9. The same A and b values are used to update all particles.

A simple but effective change stated in [18] is to replace the lines of 7-19 of Algorithm 1 with the pseudocodes given in Algorithm 2.

There are two major changes in Algorithm 2. For each particle associated part of the measurement function is linearized and A_i and b_i values are calculated using H_i matrix. Thus the calculation of the migration parameters for each particle is performed individually with varying values of A_i and b_i. The other change is that the mean estimate from the UKF/EKF is replaced with the state estimate from the Daum-Huang filter as stated on line 19 [18].

```
1    Initialization: Draw $\{x_0^i\}_{i=1}^N$ from the prior $p(x_0)$;
2    Set $\hat{x}_0$ and $m_0$ as the mean; $P_0$ as the covariance matrix.
3    for $k = 1$ to $T$ do
4        Propagate particles $x_{k-1}^i = f_k(x_{k-1}^i) + \nu_k$;
5        Calculate the mean value $\overline{x}_k$;
6        Apply UKF/EKF prediction: $(m_{k-1|k-1}, P_{k-1|k-1}) \longrightarrow (m_{k|k-1}, P_{k|k-1})$
7        for $j = 1$ to $N_\lambda$ do
8            Set $\lambda = j\Delta\lambda$;
9            Calculate $H_{\overline{x}}$ by linearizing $\gamma_k()$ at $\overline{x}_k$
10           Calculate $A$ and $b$ from (8) and (9) using $P_{k|k-1}, \overline{x}$ and $H_{\overline{x}}$;
11           for $i = 1$ to $N$ do
12               Evaluate $dx_k^i/d\lambda$ for each particle from (7)
13               Migrate particles: $x_k^i = x_k^i + \Delta\lambda.(dx_k^i/d\lambda)$
14           endfor
15           Re-evaluate $\overline{x}_k$ using the updated particles $x_k^i$.
16       endfor
17       Apply UKF/EKF update: $(m_{k|k-1}, P_{k|k-1}) \longrightarrow (m_{k|k}, P_{k|k})$;
18       Estimate $\hat{x}_k$ from the particles $x_k^i$ using $P_{k|k}$;
19       Redraw particles $x_k^i \sim N(\hat{x}_k, P_{k|k})$; (Optional)
20   endfor
```

ALGORITHM 1: Exact Flow Daum-Huang Filter.

```
7    for $j = 1$ to $N_\lambda$ do
8        Set $\lambda = j\Delta\lambda$;
9        for $i = 1$ to $N$ do
10           Calculate $H_{x_i}$ by linearizing $\gamma_k()$ at $x^i$;
11           Calculate $A^i$ and $b^i$ from (8) and (9) using $P_{k|k-1}, \overline{x}$ and $H_{x_i}$;
12           Evaluate $dx_k^i/d\lambda$ for each particle from (7)
13           Migrate particles: $x_k^i = x_k^i + \Delta\lambda.(dx_k^i/d\lambda)$
14       endfor
15       Re-evaluate $\overline{x}_k$ using the updated particles $x_k^i$.
16   endfor
17   Apply UKF/EKF update: $(m_{k|k-1}, P_{k|k-1}) \longrightarrow (m_{k|k}, P_{k|k})$;
18   Estimate $\hat{x}_k$ from the particles $x_k^i$ using $P_{k|k}$;
19   Set $m_{k|k} = \hat{x}_k$;
20   Redraw particles $x_k^i \sim N(\hat{x}_k, P_{k|k})$; (Optional)
```

ALGORITHM 2: Local Exact Flow Daum-Huang Filter.

3. Clustered Particle Flow Filter

EDH filter calculates migration parameters for all of the particles. LEDH filter calculates the migration parameters individually for each particle. CEDH filter employs clustering of the particles considering the ones producing similar errors. The calculation cost is significantly reduced through calculating common migration parameters for the particles within each cluster.

In this study, k-means++ algorithm [28] is adopted to fulfill the clustering demand of CEDH filter. The well-known k-means class methods aim at minimizing the mean square distance between points in the same set [29]. Arthur and Vassilvitskii proposed an algorithm that is $O(\log k)$-competitive. It strengthens the clustering scheme with a randomized seeding technique and improves both the speed and the accuracy in clustering.

3.1. Implementation of Clustered Particle Flow Filter. The cost of calculation is anticipated to be high in LEDH filter, since it pursues the calculation of the flow for each particle in each pseudo-time interval. The main objective of the CEDH filter is that it reduces the cost of calculation through implementing the steps of a clustering methodology. Thus, it reduces the influence of particles having a high error margin on the overall system and therefore improves the performance of the LEDH filter. The pseudocodes for the implementation of the CEDH filter are specified in Algorithm 3.

N_c is the number of clusters. Algorithm 3, like the LEDH filter, changes the steps on lines 7 to 19 of the Algorithm 1. Clustering step takes place over the error margin of the particles as expressed on line 12. The migration parameters are calculated up to the number of clusters through employing the steps depicted on lines 13 to 17. Therefore in CEDH

```
7    for j = 1 to N_λ do
8        Set λ = jΔλ;
9        for i = 1 to N do
10           Calculate H_x̄ by linearizing γ_k() at x^i;
11       endfor
12       Cluster particles over the margin of error
13       for c = 1 to N_c do
14           Calculate A^c and b^c from (6) and (7) using P_{k|k-1}, x̄ and H_{x_c};
15           Evaluate dx_k^c/dλ for each particle from (5)
16           Migrate particles: x_k^c = x_k^c + Δλ.(dx_k^c/dλ)
17       endfor
18       Eliminate the largest cluster with the biggest error
19       Re-evaluate x̄_k using the updated particles x_k^i.
20   endfor
21   Apply UKF/EKF update: (m_{k|k-1}, P_{k|k-1}) ⟶ (m_{k|k}, P_{k|k});
22   Estimate x̂_k from the particles x_k^i using P_{k|k};
23   Redraw particles x_k^i ~ N(x̂_k, P_{k|k}); (Optional)
```

ALGORITHM 3: Clustered Exact Flow Daum-Huang Filter.

filter the particles are not associated with a single parameter calculation. Also, the migration parameters are not calculated for each individual particle.

CEDH filters fight to reduce the cost of calculation in LEDH filters which typically occurs due to exhaustive calculation of each migration parameter. In addition, while reducing the cost of calculation, that the particles with high error margin can be systematically clustered together leads to significant performance improvement. The assessment of the performance and calculation speed as the outcomes of CEDH filter implementation for the scenario of multidimensional target tracking will be presented in the next section.

4. Simulation and Results

The particle filters are applied to a multitarget tracking problem which is adapted from [30]. A wireless sensor network model of 25 acoustic amplitude sensor nodes located at the intersections of the grids on a $40m \times 40m$ rectangular region is used. The model is shown in Figure 1. There are four targets moving along two axes independently. The overall state vector contains four states for these four targets. Therefore, it is in the 16-dimensional state space.

The independent movement model of four targets ($P = 4$) moving at a constant speed can be expressed as

$$x_k^{(p)} = Fx_{k-1}^{(p)} + W_p v_k^{(p)} \tag{10}$$

where $x_k^p = [x_k^p, y_k^p, \dot{x}_k^p, \dot{y}_k^p]$ consists of the x-y position and x-y velocity components of the corresponding target.

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

is the transition matrix, $v_k^{(p)} \sim N(0, \sigma_v^2 V)$ is the process noise, σ_v^2 is set to 0.01, and the covariance of the process noise for the filters is set as

$$W_p = \begin{pmatrix} 0.5 & 0 & 0.2 & 0 \\ 0 & 0.5 & 0 & 0.2 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{12}$$

All targets emit sounds with amplitude ψ and each sensor records the sum of amplitudes. Thus, measurement function for the s-th sensor located at R^s is additive:

$$\bar{z}^s(x_k) = \sum_{P=1}^{P} \frac{\psi}{\left\| \left(x_k^{(p)}, y_k^{(p)} \right)^T - R^2 \right\|_2 + d_0} \tag{13}$$

where $d_0 = 0.1$ and $\psi = 10$ in our simulations. The measurements are perturbed by Gaussian noise. There are $N_s = 25$ sensors located at grid intersections as shown in Figure 1. The four targets are initialized with states [8 8 0.01−0.01], [9 34−0.001 0.003], [34 32 0.02 − 0.1], and [35 12 0.001 − 0.001], respectively.

Multitarget tracking scenario associated with the wireless sensor network model was simulated employing the filters of EKF, BPF, EDH, LEDH, and CEDH for different measurement sets. Each run starts with a different initial distribution. Figure 1 shows the true routes of the targets and the estimated routes obtained with CEDH. The simulation consists of 40 time steps and each algorithm is run separately 100 times. The optimal mass transfer (OMAT) metric is calculated by taking the mean of each time interval. The average OMAT error calculated for EKF, BPF, EDH, LEDH, and CEDH is given in Figure 2.

As shown in Figure 2, the EKF filter has the highest error margin. It is followed by the BPF, EDH, and LEDH, respectively. CEDH filters with different cluster numbers are the best performing filters at all time intervals.
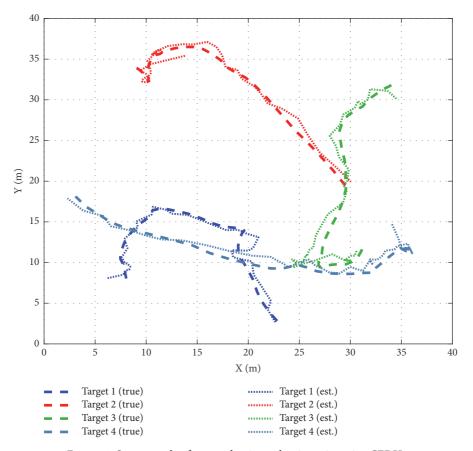
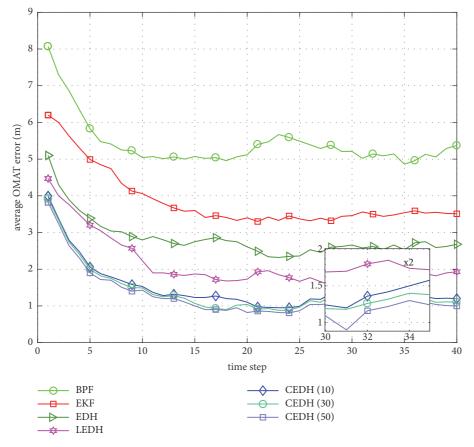FIGURE 1: One example of true and estimated trajectories using CEDH.

FIGURE 2: Average OMAT errors of the filters.

TABLE 1: Detailed information about the filters.

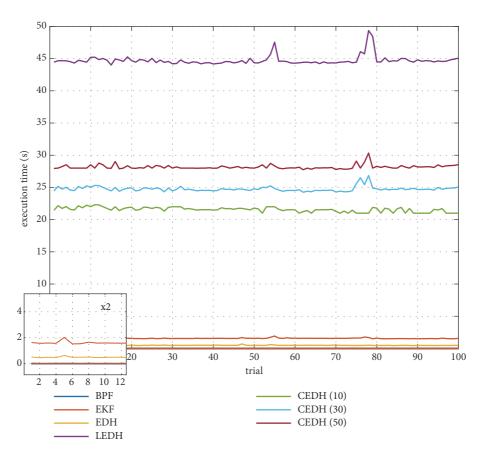| Algorithm	EKF	BPF	EDH	LEDH	CEDH	CEDH	CEDH
Particle num.	N/A	10000	500	500	500	500	500
λ	N/A	N/A	30	30	30	30	30
Cluster num.	N/A	N/A	N/A	N/A	10	30	50
Avg. err. (m)	4,47	3,79	3,24	2,60	1,50	1,41	1,37
Exec. time (s)	0,03	1,57	0,48	44,73	23,05	24,79	28,42

FIGURE 3: Execution times of the filters.

Detailed information about the filters is given in Table 1. The computer used for the simulations is MacBook Air (Early 2015), which has a 1.6 GHz Intel Core i5 processor, 4 GB of 1600 MHz DDR3 memory, and an Intel HD Graphics 6000 1536 MB graphics card.

As shown in Figure 3, the CEDH algorithm ensured significant improvement in terms of calculation cost. It corresponds to approximately half of the execution time of the LEDH algorithm. Increasing the number of clusters improves performance; however this improvement brings extra cost of calculation to the filter.

5. Conclusions

We introduced Clustered Exact Flow Daum-Huang Particle Filter in this study. While improving the performance of the LEDH filter, CEDH also reduces the cost of calculation through utilizing a clustering strategy. The consideration of clustering of the particles with high error margins ensures that the influence of the particles on the overall system can be reduced. In CEDH filter, the particles are not accompanied with the calculation of same parameter as it occurs in EDH filter. CEDH filter succeeds in reducing the cost of calculation in LEDH filter arising due to comprehensive calculation of each migration parameter. The calculation cost is significantly reduced by calculating common migration parameters for the particles with similar error values. The given multidimensional target tracking problem demonstrates the success of the CEDH filter. In the future work, the effect of clustering on the system will be examined and the focus will be on the selection of the appropriate number of clusters and elimination of the appropriate cluster with the help of machine learning.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[2] M. West and J. Harrison, *Bayesian Forecasting and Dynamic Models*, Springer Series in Statistics, New York, NY, USA, 2nd edition, 1997.

[3] G. Welch and G. Bishop, "An introduction to the Kalman filter," Technical report: UNC-CH Computer Science Technical Report 95041, 1995.

[4] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian Bayesian state estimation," *IEE proceedings F*, vol. 140, no. 2, pp. 107–113, 1993.

[5] P. Bunch and S. Godsill, "Approximations of the optimal importance density using Gaussian particle flow importance sampling," *Journal of the American Statistical Association*, vol. 111, no. 514, pp. 748–762, 2016.

[6] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[7] J. Cornebise, E. Moulines, and J. Olsson, "Adaptive methods for sequential importance sampling with application to state space models," *Statistics and Computing*, vol. 18, no. 4, pp. 461–480, 2008.

[8] M. K. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.

[9] A. Doucet, N. de Freitas, N. Gordon, and S. J. Russell, "Rao-Blackwellised particle filtering for dynamic Bayesian networks," in *Proceedings of the Uncertainty in Artificial Intelligence (UAI)*, pp. 176–183, Springer, San Francisco, Calif, USA, 2000.

[10] R. van der Merwe, A. Doucet, N. De Freitas, and E. Wan, "The unscented particle filter," in *Proceedings of the Neural Info. Proc. Sys. (NIPS)*, pp. 584–590, Denver, Co, USA.

[11] A. Beskos, D. Crisan, and A. Jasra, "On the stability of sequential Monte Carlo methods in high dimensions," *The Annals of Applied Probability*, vol. 24, no. 4, pp. 1396–1445, 2014.

[12] D. Crisan and B. Rozovskii, *The Oxford Handbook of Nonlinear Filtering*, Oxford University Press, 2011.

[13] P. Bui Quang, C. Musso, and F. Le Gland, "An insight into the issue of dimensionality in particle filtering," in *Proceedings of the 2010 13th International Conference on Information Fusion (FUSION 2010)*, pp. 1–8, Edinburgh, July 2010.

[14] M. Ades and P. J. van Leeuwen, "The equivalent-weights particle filter in a high-dimensional system," *Quarterly Journal of the Royal Meteorological Society*, vol. 141, no. 687, pp. 484–503, 2015.

[15] P. M. Djuric, T. Lu, and M. F. Bugallo, "Multiple particle filtering," in *Proc. Intl. Conf. Acoustics, Speech and Signal Proc. (ICASSP)*, vol. 3, pp. 1181–1184, 2007.

[16] A. Beskos, D. Crisan, A. Jasra, K. Kamatani, and Y. Zhou, "A stable particle filter for a class of high-dimensional state-space models," *Advances in Applied Probability*, vol. 49, no. 1, pp. 24–48, 2017.

[17] F. Septier, S. K. Pang, A. Carmi, and S. Godsill, "On MCMC-based particle methods for Bayesian filtering: Application to multitarget tracking," in *Proceedings of the 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2009)*, pp. 360–363, Aruba, Dutch Antilles, December 2009.

[18] T. Ding and M. J. Coates, "Implementation of the Daum-Huang exact- flow particle filter," in *Proceedings of the 2012 IEEE Statistical Signal Processing Workshop, SSP*, pp. 257–260, USA, August 2012.

[19] F. Daum and J. Huang, "Nonlinear filters with log-homotopy," in *Proc. SPIE Conf. Signal and Data Processing of Small Targets*, San Diego, CA, USA, 2007.

[20] F. Daum, I. Kadar, and J. Huang, "Nonlinear filters with particle flow induced by log-homotopy," in *Proceedings of the SPIE Defense, Security, and Sensing*, p. 733603, Orlando, Florida, USA.

[21] F. Daum, I. Kadar, J. Huang, and A. Noushin, "Exact particle flow for nonlinear filters," in *Proceedings of the SPIE Defense, Security, and Sensing*, p. 769704, Orlando, Florida.

[22] F. Daum, J. Huang, and A. Noushin, "Coulombs law particle flow for nonlinear filters," in *Proc. SPIE Conf. Signal and Data Proc*, San Diego, CA, USA, 2011.

[23] Y. Li, L. Zhao, and M. Coates, "Particle flow for particle filtering," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 20–25, Shanghai, China, March 2016.

[24] Y. Li and M. Coates, "Fast Particle Flow Particle Filters via Clustering," in *Proceedings of the 19th International Conference on Information Fusion*, Heidelberg, Germany, 2016.

[25] S. C. Surace, A. Kutschireiter, and J.-P. Pfister, "How to avoid the curse of dimensionality: scalability of particle filters with and without importance weights," https://arxiv.org/abs/1703.07879.

[26] F. Daum, J. Huang, and A. Noushin, "New Theory and Numerical Results for Gromov's Method for Stochastic Particle Flow Filters," in *Proceedings of the 21st International Conference on Information Fusion, FUSION 2018*, pp. 108–115, UK, July 2018.

[27] Y. Li, S. Pal, and M. Coates, "Invertible Particle Flow-based Sequential MCMC with ex- tension to Gaussian Mixture noise models," *IEEE Transactions on Signal Processing*, 2019.

[28] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027–1035, 2007.

[29] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[30] O. Hlinka, O. Sluciak, F. Hlawatsch, P. M. Djuric, and M. Rupp, "Distributed Gaussian particle filtering using likelihood consensus," in *Proceedings of the ICASSP 2011 - 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3756–3759, Prague, Czech Republic, May 2011.